Ответы к экзамену по МИПиС

Моделирование информационных процессов и систем

Единый конспект по вопросам

Содержание

Основы моделирования

1. Моделирование как способ априорной оценки эксплуатационных характеристик разрабатываемых информационных систем. Основные эксплуатационные характеристики информационных систем

Моделирование используют для априорной оценки эксплуатационных характеристик ИС, потому что до завершения разработки еще нельзя полноценно проверить будущую систему в реальной эксплуатации. Вместо этого строят модель или прототип, проводят эксперименты и заранее оценивают, как проектные решения повлияют на работу системы.

В лекции эта идея связана с эволюционным прототипированием: модель ИС постепенно уточняется от формального описания до тестируемого макета, а на каждом этапе помогает оценивать характеристики будущей системы и корректировать структуру, программные и технические решения.

К основным эксплуатационным характеристикам ИС относятся:

- производительность - пропускная способность, скорость обработки запросов и время отклика;
- надежность - устойчивость к сбоям, отказоустойчивость и время восстановления после сбоя;
- доступность - доля времени, когда система доступна пользователям, а также влияние планового и внепланового обслуживания;
- масштабируемость - способность увеличивать производительность за счет добавления узлов или усиления существующих ресурсов;
- экономическая эффективность - стоимость владения и соотношение затрат с достигаемыми целями.

Для информационных систем, которые удобно рассматривать как системы обслуживания, эти характеристики конкретизируются через среднее время обслуживания, длину очереди, загрузку каналов, вероятность отказа, абсолютную и относительную пропускную способность. Поэтому модель позволяет заранее увидеть, где возникнут очереди, перегрузка, потери заявок или недоиспользование ресурсов.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 5-7; `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 7-9; `ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf`, с. 44-49.

2. Научные основы моделирования: понятие аналогии и подобия сущностей, виды подобия

Аналогия, или подобие, - фундаментальная основа научного моделирования. В лекции она определяется как эквивалентность разных сущностей по выделенным общим свойствам или поведению. Если такие свойства для цели исследования совпадают, модель можно использовать как заменитель оригинала и переносить результаты анализа модели на предмет моделирования.

Аналогия бывает качественной и количественной. Качественная аналогия опирается на признаки, которые трудно или невозможно строго измерить: удобство, похожесть, воспринимаемая форма. Количественная аналогия выражается через измеримые свойства и поэтому важнее для научного моделирования.

В лекционном материале выделены три вида количественного подобия:

- физическое подобие - разные материальные объекты проявляют одинаковые физические свойства и могут практически заменять друг друга в физическом процессе;
- математическое подобие - разные объекты описываются одинаковыми математическими структурами, например одной и той же системой уравнений;
- виртуальное, или поведенческое, подобие - структура и поведение объекта имитируются компьютерными средствами, а модель воспроизводит существенную логику работы оригинала.

Для информационных систем особенно важно виртуальное подобие: прототип может имитировать процессы, интерфейсы, обмен сообщениями и даже работу еще не реализованных функций с помощью заглушек. При этом модель не обязана копировать все детали оригинала. Она должна сохранять именно те свойства, по которым оцениваются эксплуатационные характеристики.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 8-9.

3. Цель и состав процесса моделирования

Цель моделирования - получить средство исследования предмета моделирования, которое позволяет понять его поведение, спрогнозировать результаты при разных условиях и обосновать проектные решения. Иначе говоря, модель строится ради ответа на конкретные вопросы о системе, а не ради самой модели.

По материалам курса процесс моделирования включает несколько последовательных этапов. Сначала определяются объект и предмет моделирования, затем формулируется цель исследования. После этого строится словесное описание предмета, выделяются параметры и отношения между ними, выбирается модель времени и выполняется формализация. Далее создается вычислительная модель, проводятся эксперименты, проверяется адекватность и интерпретируются полученные результаты.

В сжатом виде состав процесса моделирования можно представить так:

1. Выделение объекта, предмета и цели моделирования.
2. Построение семантической, то есть словесной, модели.
3. Анализ параметров и отношений между ними.
4. Выбор модели времени и построение формальной модели.
5. Построение вычислительной модели.
6. Проведение экспериментов с моделью.
7. Проверка адекватности и анализ результатов.

Именно такая схема проходит через лабораторные работы курса: от описания объекта и предмета моделирования до экспериментов в AnyLogic или расчетов по конечно-разностной схеме.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 10-13.

4. Понятие эволюционного прототипирования, моделирование как основа и средство эволюционного прототипирования разрабатываемой информационной системы

Эволюционное прототипирование - это разработка информационной системы через последовательно уточняемую модель, или прототип, на котором заранее оценивают эксплуатационные характеристики проектных решений. В лекции прямо подчеркивается: характеристики будущей ИС зависят от структуры, технических и программных средств, поэтому их нужно анализировать до завершения разработки.

Эволюционирование прототипа может начинаться с формального описания, например UML-модели, а затем переходить к масштабируемому, модифицируемому и периодически тестируемому натурному макету системы. Такой макет постепенно приближается к окончательному виду реальной ИС.

Моделирование здесь выступает не вспомогательной картинкой, а основным средством прототипирования. Модель имитирует структуру и работу всей системы или ее еще не реализованных частей, позволяет оценивать время отклика, надежность, доступность, масштабируемость и другие характеристики, а затем оптимизировать структуру ИС и реализуемые в ней информационные процессы.

Главный смысл подхода - начинать анализ характеристик на ранних стадиях разработки и продолжать его по мере уточнения прототипа. Поэтому модель в эволюционном прототипировании живет вместе с проектом: она проверяется, усложняется, масштабируется и используется для принятия проектных решений.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 5-7.

5. Семантические словесные модели - их отличие от формальных моделей

Семантическая, или словесная, модель - это содержательное описание предмета моделирования на естественном языке. Она отвечает на вопросы, что представляет собой объект, какие в нем есть элементы, какие процессы существенны и каков общий смысл происходящего.

Формальная модель - это описание того же предмета в точных терминах параметров, отношений, формул, логических правил, таблиц или алгоритмов. Она нужна тогда, когда поведение объекта уже требуется не только объяснить, но и вычислять, проверять и исследовать в эксперименте.

В материалах курса этот переход показан на примере пружинного маятника. Сначала предмет моделирования описывается словами: есть груз, пружина, жидкая среда и затухающие колебания. Затем эта словесная модель интерпретируется в терминах параметров, выделяются внутренние и внешние характеристики, вводится время и записываются дифференциальные уравнения. Так словесное описание превращается в формальную модель.

Главное отличие состоит в следующем:

- семантическая модель дает содержательное понимание предмета;
- формальная модель задает точные зависимости между параметрами;
- семантическая модель удобна для первичного анализа;
- формальная модель необходима для вычислений и экспериментов.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 11-13; `ЛР №1 Моделирование динамических систем в AnyLogic.pdf`, с. 4-8.

6. Схемы моделирования, их роль в построении формальных моделей. Обобщенная схема моделирования

Схема моделирования - это общий способ формального представления предмета моделирования через параметры, их классификацию и отношения между ними. Она задает каркас будущей формализации и позволяет переходить от словесного описания объекта к строгой модели не произвольно, а по определенному порядку.

Роль схем моделирования состоит в том, что они помогают системно выделить:

- какие параметры являются внутренними, а какие внешними;
- что выступает входным воздействием, а что выходной характеристикой;
- какие параметры управляемые, а какие нет;
- какие параметры экзогенные, а какие эндогенные;
- какие из них являются константами, статическими или динамическими;
- каким образом формируются их значения - задаются напрямую или вычисляются по формуле.

В курсе эта логика называется обобщенной схемой моделирования. Она особенно ясно показана в работе по движению шара, где для каждого параметра строится таблица с его характеристиками, после чего выбирается модель времени и задаются отношения между параметрами в форме дифференциальных уравнений. То есть обобщенная схема моделирования - это порядок перехода от анализа предмета к построению формальной математической и затем вычислительной модели.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 11-13, 31-38; `ЛР №1 Моделирование динамических систем в AnyLogic.pdf`, с. 5-9; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 6.

7. Адекватность модели предмету моделирования. Проблемы оценки адекватности модели

Адекватность модели - это ее соответствие предмету моделирования в тех свойствах, которые существенны для цели исследования. Адекватная модель не обязана воспроизводить объект во всех деталях. Достаточно, чтобы она правильно отражала те закономерности, ради которых модель строилась.

В курсе адекватность проверяется через эксперимент и сопоставление с известными свойствами исследуемого процесса. В модели движения шара сравнивают результаты с известными законами кинематики и действием силы Архимеда. В модели банковского отделения проверяют, выполняются ли ожидаемые свойства системы обслуживания: при нулевой интенсивности потока очереди отсутствуют, абсолютная пропускная способность равна нулю, относительная равна единице, вероятность отказа равна нулю; при увеличении нагрузки растут очереди, появляется отказ и меняются значения пропускной способности.

Оценка адекватности сложна по нескольким причинам. Во-первых, она всегда зависит от цели моделирования: модель может быть адекватной для оценки пропускной способности, но недостаточной для оценки надежности или стоимости системы. Во-вторых, любая модель строится с упрощениями, поэтому часть факторов неизбежно исключается. В-третьих, на результат влияют точность исходных данных, выбранные допущения, чувствительность модели к параметрам и численные погрешности вычислений. Для непрерывных моделей дополнительно важны шаг дискретизации и устойчивость метода. Поэтому адекватность на практике оценивают как достаточную репрезентативность результатов модели по отношению к реальному предмету моделирования.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 9-10, 39-41; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 5-6; `ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf`, с. 47-50.

Обобщенное агрегатное моделирование

8. Основные понятия агрегатного моделирования: агрегатная модель, состояние агрегата, взаимодействие агрегатов

В агрегатном моделировании систему рассматривают как совокупность относительно самостоятельных частей - агрегатов. Каждый агрегат представляется как обособленный фрагмент модели, имеющий собственные параметры, внутреннюю логику функционирования, входы и выходы. Такой подход удобен потому, что сложную систему можно разложить на более простые подсистемы и исследовать их связи.

Агрегатная модель - это модель, в которой объект описывается не как неделимое целое, а как набор взаимодействующих агрегатов. Состояние агрегата - это совокупность значений его существенных внутренних параметров в данный момент времени. Если эти параметры меняются, то меняется и состояние агрегата. В непрерывной модели состояние может задаваться, например, координатой и скоростью, а в дискретной - текущим режимом работы и содержимым памяти.

Взаимодействие агрегатов происходит через обмен входными и выходными воздействиями, сообщениями, ресурсами или значениями параметров. Один агрегат формирует выход, который становится внешним воздействием для другого. Поэтому поведение системы в целом определяется не только свойствами отдельных агрегатов, но и структурой их связей.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 14-16.

9. Схема обобщенного моделирования простого агрегата

Схема обобщенного моделирования простого агрегата задает порядок его формального описания. Сначала выделяют сам агрегат как объект моделирования, затем определяют цель исследования, после чего перечисляют существенные параметры и устанавливают отношения между ними. Далее параметры классифицируют по роли в модели, выбирают модель времени и только после этого переходят к математической или алгоритмической записи поведения агрегата.

Для простого агрегата обычно фиксируют:

- внутренние параметры, описывающие его собственное состояние;
- внешние параметры, задающие влияние среды;
- входные воздействия;
- выходные характеристики;
- правила изменения состояния и формирования выходов;
- модель времени, в которой развивается поведение агрегата.

Именно такой порядок в курсе используется при переходе от словесного описания предмета к формальной модели. Сначала задается множество параметров, затем определяется их роль, а после этого отношения между ними интерпретируются, например, как система дифференциальных уравнений, таблица переходов или алгоритм.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 17-19.

10. Классы параметров простого агрегата: внутренние, внешние, входные, выходные

При обобщенном описании простого агрегата параметры делят по их смысловой роли в модели.

Внутренние параметры характеризуют сам агрегат и его собственные свойства. Они описывают то, что принадлежит агрегату как объекту. Например, масса тела или жесткость пружины в физической системе являются внутренними параметрами.

Внешние параметры относятся к среде, в которой работает агрегат. Они не принадлежат агрегату, но влияют на его поведение. К таким параметрам можно отнести свойства окружающей среды, внешнюю нагрузку, температуру, интенсивность входного потока.

Входные параметры выражают воздействия на агрегат. Это те значения, которые поступают в агрегат извне и запускают или направляют его поведение. Они часто совпадают с управляемыми экзогенными параметрами.

Выходные параметры являются результатом функционирования агрегата. По ним оценивают его состояние и качество работы. В курсовых примерах выходами выступают положение и скорость тела, длина очереди, пропускная способность, вероятность отказа.

Один и тот же параметр может одновременно принадлежать нескольким классификационным группам. Например, входное воздействие обычно является внешним по происхождению, а выходная характеристика - нередко эндогенной по способу формирования.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 17-19.

11. Анализ и интерпретация параметров простого агрегата А

Анализ параметров простого агрегата состоит в том, чтобы не просто перечислить величины, связанные с объектом, а для каждой из них установить роль в модели. Нужно определить, является ли параметр внутренним или внешним, входным или выходным, управляемым или нет, экзогенным или эндогенным, статическим или динамическим, а также понять, как он формируется.

Интерпретация параметров означает перевод содержательного описания агрегата в формальные термины. Иначе говоря, параметр надо связать с конкретным физическим, информационным или логическим смыслом и указать, каким образом его значение участвует в поведении модели. В курсе это делается через таблицу характеристик параметров и через последующую запись отношений между ними.

Например, в модели пружинного маятника смещение и скорость интерпретируются как выходные динамические параметры, коэффициент жесткости - как внутренняя константа, коэффициент сопротивления среды - как внешний фактор, а начальное смещение - как управляемое входное воздействие. После такой интерпретации уже можно выбрать подходящую математическую схему модели.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 19-20.

12. Формирование значений параметров агрегата А

При построении агрегатной модели каждому параметру нужно задать множество допустимых значений и способ их получения. Значения могут быть количественными или качественными, непрерывными или дискретными. Для количественного параметра дополнительно задают систему измерения, единицу и область возможных значений Ω_p.

По способу получения значений параметры делятся прежде всего на детерминированные и случайные. Детерминированный параметр получает значение однозначно, без учета случайного фактора. Случайный параметр получает значение апостериорно, то есть в ходе моделирования, и результат заранее не известен.

Случайные параметры в лекции дополнительно делятся на стохастические и спорадические. Стохастический параметр имеет заданный закон распределения вероятностей, поэтому его можно моделировать генератором случайных чисел. Спорадический параметр связан с непредсказуемыми событиями: авариями, изменением режима работы, редкими внешними воздействиями. Если такой параметр нужно формально исследовать, его либо описывают правилом возникновения события, либо приближают стохастической моделью.

В итоге для каждого параметра агрегата фиксируют не только имя и смысл, но и то, как он будет получать значения: задаваться как константа, устанавливаться перед экспериментом, меняться по известному закону, вычисляться из других параметров или генерироваться случайно.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 20-21.

13. Категории параметров агрегата A: экзогенные, эндогенные и экстернальные параметры

Экзогенные параметры - это независимые параметры модели. Их значения задаются непосредственно: исходными данными, условиями эксперимента, внешней средой, входными воздействиями или заранее выбранным законом изменения во времени. В качестве экзогенных могут выступать входные воздействия X, внешние параметры V, внутренние параметры H и особый параметр времени t.

Эндогенные параметры - зависимые параметры. Их значения определяются самой моделью через отношения с другими параметрами. Например, выходные характеристики системы обычно являются эндогенными, потому что вычисляются по входам, внутренним параметрам, внешним воздействиям и текущему состоянию.

Экстернальные параметры в лекции выделяются как внешние характеристики агрегата. Это параметры, через которые агрегат наблюдается или взаимодействует с окружением. Противоположная категория - интернальные параметры: они используются для описания внутреннего состояния агрегата и не рассматриваются как внешние характеристики.

Важно не смешивать эти классификации. Экзогенность и эндогенность говорят о способе получения значения, а интернальность и экстернальность - о роли параметра относительно агрегата: внутреннее состояние или внешняя характеристика.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 21-22.

14. Управляемые, неуправляемые и устанавливаемые параметры простого агрегата A

В обобщенной схеме моделирования управляемость задают для экзогенных параметров агрегата. Управляемый параметр - это параметр, значение которого субъект моделирования может варьировать во время эксперимента. Например, можно менять интенсивность потока заявок, чтобы посмотреть, как система выдерживает нагрузку.

Устанавливаемый параметр задается до начала эксперимента и далее в ходе прогона вручную не корректируется. Так обычно задают объем памяти, пропускную способность канала, число ресурсов, начальную температуру или интенсивность помех. Такой параметр описывает режим эксперимента.

Неуправляемый параметр доступен только для наблюдения. Исследователь не может или не должен менять его значение. Это могут быть свойства внешней среды, физические характеристики объекта, заданная производительность готовой вычислительной системы, масса, плотность, атмосферное давление и похожие факторы.

Разделение нужно для постановки эксперимента. Управляемые параметры используют как факторы воздействия, устанавливаемые - как настройки варианта модели, а неуправляемые - как ограничения и наблюдаемые условия.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 22-23.

15. Параметр времени. Модели времени: непрерывное, реальное, виртуальное время

Параметр времени нужен для упорядочивания сменяющихся состояний модели. В лекции время не входит в множество параметров самого агрегата, а считается выходной характеристикой отдельной модели времени M_T. Обращение к функции $time()$ возвращает текущее значение времени, по которому упорядочиваются состояния агрегата.

Модель непрерывного времени $M_T(t_0, \delta t)$ задает абстрактную непрерывную шкалу, где тик является бесконечно малой величиной. Такая модель нужна для дифференциального описания динамических параметров, например температуры или координаты физического объекта.

Модель реального времени $M_T(t_0, \Delta t)$ задает дискретную шкалу с конечным тиком Δt. Текущее значение времени меняется скачком после истечения заданного интервала. На практике длительность тика может быть связана с физическими часами компьютера и изменяться для ускорения или замедления эксперимента.

Модель виртуального времени $M_T(t_0, \sim)$ используется, когда важен не физический интервал между состояниями, а их порядок. При каждом спорадическом переходе состояние просто получает следующий номер, обычно $t = 0, 1, 2, \ldots$. Поэтому виртуальное время удобно для автоматов и дискретно-событийных моделей, где такт срабатывания имеет неопределенную длительность.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 23-27.

16. Динамические и статические параметры простого агрегата A

Динамический параметр - это параметр, значение которого вычисляется для каждого текущего значения времени: $d(t) = F(t, \ldots)$. Если эндогенный параметр зависит хотя бы от одного динамического параметра, он тоже становится динамическим, потому что его значение опосредованно меняется во времени.

Статический параметр не является функцией текущего времени. При этом он не обязательно вечная константа: в лекции статический параметр может спорадически изменяться по некоторому условию, но не вычисляется как непрерывная функция времени. Пример - параметр, задающий режим выполнения модели.

Отдельно выделяются константы. Константа в течение текущего эксперимента не меняет заданного значения. Она может быть абсолютной, как физическая постоянная, или установленной перед прогоном, как объем памяти, пропускная способность канала или производительность процессора.

На экзамене важно сказать коротко: динамический параметр связан с текущим временем, статический - с состоянием или условием без явной временной функции, константа - с неизменным значением в пределах эксперимента.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 28-29.

17. Динамические, статические и стационарные модели

Динамической называют модель, среди экстернальных параметров которой есть динамические выходные характеристики. Такая модель описывает изменение состояния во времени и позволяет строить траектории поведения объекта.

Статическая модель имеет только статические экстернальные параметры. Она может описывать последовательную смену значений в виртуальном времени, но ее выходные характеристики не являются функциями непрерывного времени. Поэтому статическая модель обычно фиксирует режим, состояние или логическую зависимость без непрерывной динамики.

Стационарность - это свойство динамической модели, а не просто синоним статичности. Если при росте времени экстернальные динамические параметры не стремятся к устойчивым постоянным значениям, модель считается нестационарной. Если с некоторого момента значения выходных динамических параметров стабилизируются и далее не меняются с заданной точностью, модель находится в стационарном состоянии.

В лабораторной задаче о пластине стационарное состояние видно по температурным слоям: очередная матрица температур практически не отличается от предыдущей. Тогда переходный процесс завершен, и дальнейшее моделирование уже не дает новых существенных значений.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 29-33; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 31-34.

18. Обобщенное моделирование сложных систем

Обобщенное моделирование сложной системы начинается не с попытки описать все одним большим агентом, а с разбиения объекта на взаимодействующие подсистемы. Для каждой подсистемы задают назначение, внутреннее состояние, входы, выходы и связи с другими подсистемами. Затем такую структуру уточняют до уровня, где отдельные части уже можно описать базовыми средствами моделирования.

В иерархической модели каждая подсистема может быть представлена отдельным агентом или агрегатом. Агент инкапсулирует собственные внутренние элементы, параметры и поведение, а наружу выводит только интерфейс - переменные, входные воздействия и результаты, через которые он взаимодействует с другими агентами. Это снижает сложность: отдельные части можно разрабатывать и проверять отдельно, а затем собирать в единую модель.

Для сложных информационных систем такой подход особенно важен. В них одновременно существуют процессы обслуживания, очереди, ресурсы, управляющие правила, обмен данными и внешняя среда. Обобщенная модель связывает эти части в одну архитектуру, после чего для отдельных отношений выбирают конкретные схемы: функциональные, автоматные, стохастические, процессные или гибридные.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 42-45; `Модели с иерархической структурой.docx`, с. 1-2.

Моделирование отношений

19. Интерпретация обобщенной модели простого агрегата. Математические схемы. Формы интерпретации отношений над параметрами модели и способы их выражения

Интерпретация обобщенной модели простого агрегата нужна потому, что после выделения параметров модель еще слишком абстрактна. В ней указано, какие параметры есть и какие сигнатуры отношений их связывают, но не задано, как именно получать значения экзогенных параметров и вычислять эндогенные. Интерпретация конкретизирует эти процедуры.

Экзогенные параметры являются первичными: для них задают способ формирования значений во времени. Эндогенные параметры получают значения через отношения с другими параметрами, поэтому для них нужно конкретизировать каждую сигнатуру отношения $r(p_1, p_2, \dots, p_n)$.

Для такой конкретизации используются математические схемы. Математическая схема - это конкретный математический аппарат или метод, с помощью которого параметры и отношения обобщенной модели превращаются в модель, пригодную для экспериментов и анализа. В курсе выделяются типовые схемы: D-схемы, F-схемы, P-схемы и Q-схемы.

Отношения над параметрами могут иметь функциональную или автоматную форму. Выражаются они аналитически, графически, таблично или алгоритмически. Поэтому одна и та же содержательная связь может быть записана формулой, системой уравнений, таблицей переходов, графом состояний, рекуррентным алгоритмом или программной процедурой.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 31-38.

20. Общий вид функциональной модели

Функциональная модель возникает тогда, когда отношения между параметрами агрегата выражаются как зависимость вектора выходных характеристик от остальных параметров. Для динамической или статической модели это записывается в виде

$$\vec{y}_A(t) = \vec{F}_A(\vec{x}, \vec{v}, \vec{h}, t),$$

где $\vec{x}$ - входные воздействия, $\vec{v}$ - внешние факторы, $\vec{h}$ - внутренние параметры, t - время, а $\vec{F}_A$ - абстрактный векторный оператор.

Оператор $\vec{F}_A$ в лекции называется законом функционирования агрегата. Зависимость выходных характеристик от времени образует выходную траекторию агрегата.

Если агрегат находится в стационарном состоянии, параметр времени из отношения исключается. Тогда функциональная модель задается отображением

$$\vec{y}_A = \vec{f}_A(\vec{x}, \vec{v}, \vec{h}).$$

Итак, функциональная модель удобна там, где выходы можно выразить как результат преобразования входов, внешних факторов и внутренних параметров без отдельного описания памяти в виде автомата.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 34.

21. Общий вид автоматной модели

Автоматная модель нужна тогда, когда выходные характеристики зависят не только от текущих значений параметров, но и от предыстории. Чтобы учесть эту предысторию, вводят состояние агрегата $\vec{Z}_A(t)$, а функционирование рассматривают как последовательную смену состояний во времени.

Состояние агрегата в момент времени можно представить как вектор значений всех параметров:

$$\vec{Z}_A(t) = \langle p_1(t), p_2(t), \dots, p_N(t) \rangle.$$

Тогда поведение агрегата описывается фазовой траекторией в пространстве состояний. Текущее состояние вычисляется по начальному состоянию, входам, внешним и внутренним параметрам, выходам и времени:

$$\vec{Z}_A(t) = \Phi(\vec{Z}_A(t_0), \vec{x}, \vec{v}, \vec{h}, \vec{y}, t).$$

Выходные характеристики затем формируются оператором преобразования текущего состояния:

$$\vec{y}_A = \vec{F}_Z(\vec{Z}_A(t)).$$

Такая агрегатная модель называется автоматной: она описывает не просто функциональную зависимость выхода, а последовательность переходов между состояниями и выходы, порождаемые текущим состоянием.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 35-36; `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 2-7.

Эксперименты с моделью

22. Формы выражения экспериментов с аналитическими и вычислительными моделями

Эксперимент с аналитической моделью обычно выражается как теоретическое исследование формул, уравнений и их следствий. В этом случае меняют параметры, анализируют решение, находят экстремумы, устойчивые режимы, предельные значения, строят графики зависимостей. То есть эксперимент проводится прежде всего как рассуждение и вычисление над математическим описанием.

Эксперимент с вычислительной моделью выражается в серии прогонов модели при заданных исходных данных и настройках. Здесь результаты получают не из готовой формулы, а путем последовательного вычисления состояний модели во времени или по событиям. Поэтому для вычислительного эксперимента важны сценарий запуска, шаг времени, продолжительность прогона, собираемая статистика и число повторений.

На практике аналитические и вычислительные формы часто сочетаются. Сначала аналитическая модель помогает понять ожидаемые свойства системы, а затем вычислительный эксперимент позволяет проверить адекватность, исследовать сложные режимы и оценить характеристики, которые трудно выразить в замкнутой формуле.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 38-39.

23. Виды экспериментов для оценки свойств агрегатной модели: валидация, калибровка, чувствительность, масштабирование

Валидация - это проверка того, насколько модель адекватно отражает предмет моделирования в рамках поставленной цели. В курсе такая проверка строится через сопоставление результатов модели с известными теоретическими свойствами системы или с ожидаемым поведением объекта.

Калибровка - это подбор значений параметров модели так, чтобы ее результаты лучше согласовывались с наблюдаемыми или нормативными данными. По сути, это настройка модели.

Анализ чувствительности показывает, насколько сильно результаты зависят от изменения входных параметров. Если небольшое изменение параметра заметно влияет на выход, значит модель чувствительна к этому параметру.

Масштабирование - это исследование поведения модели при изменении масштабов системы: числа агентов, интенсивности потоков, размеров объекта, временного шага, количества ресурсов. Такой эксперимент позволяет понять, как изменяются характеристики при росте или уменьшении системы.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 39-41.

24. Оптимизационные эксперименты с моделями

Оптимизационный эксперимент предназначен для автоматического поиска таких значений управляемых параметров модели, при которых целевая функция принимает наилучшее значение. Это может быть минимум времени обслуживания, минимум вероятности отказа, максимум дальности полета, максимум пропускной способности и так далее.

По сравнению с простым прогоном здесь недостаточно просто наблюдать поведение модели. Нужно задать изменяемые параметры, ограничения на их значения, целевой критерий и способ сравнения вариантов. Далее система многократно запускает модель, перебирая или интеллектуально подбирая значения параметров и оценивая результат.

В курсе оптимизационный эксперимент связывается с поиском таких параметров модели, которые обеспечивают требуемое поведение объекта. Поэтому оптимизация - это не отдельная теория от модели, а способ использовать модель как инструмент выбора наилучшего решения.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 40; `Тема_7_Имитационное_моделирование.pdf`, с. 10.

25. Эксперименты по сценариям

Сценарный эксперимент - это исследование модели при заранее подготовленных наборах условий, каждый из которых отражает отдельную возможную ситуацию функционирования системы. Сценарий задает комбинацию параметров, внешних воздействий, ограничений и, иногда, правил поведения среды.

Смысл такого эксперимента в сравнении вариантов. Например, можно рассмотреть нормальную нагрузку, пиковую нагрузку, отказ части ресурсов, изменение интенсивности потока, ускоренный темп обслуживания или другой шаг дискретизации. Для каждого сценария выполняется прогон модели, после чего сравниваются выходные характеристики.

Сценарный подход удобен тем, что он связывает модель с практическими вопросами: что будет, если условия изменятся именно так. Поэтому он особенно полезен в проектировании и подготовке управленческих решений.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 40.

Моделирование сложных систем

26. Понятие сложной системы. Свойства сложных систем. Структурное моделирование сложных систем

Сложная система - это система, которую нельзя адекватно представить как один простой агрегат с набором параметров и отношений. Для информационных систем это особенно типично: они состоят из множества объектов, процессов, каналов связи, накопителей, ресурсов и управляющих правил, работающих параллельно и влияющих друг на друга.

В лекции выделены три дополнительных системных свойства сложных систем:

- структурность - система рассматривается как целостность взаимосвязанных и взаимодействующих информационных объектов;
- иерархичность - объекты верхних уровней раскрываются через структуры взаимосвязанных объектов нижних уровней;
- сложная функциональность - система в целом и каждый объект на своем уровне имеют локальную цель, параметры, выходные характеристики и отношения между ними.

Структурное моделирование начинается с выделения объектов, уровней и связей между ними. Если объект текущего уровня сам сложен, его декомпозируют дальше. Так продолжают до уровня простых агрегатов, для которых уже можно применять обобщенную схему моделирования простого агрегата.

Именно поэтому структурная часть модели не является второстепенной. Без нее нельзя корректно описать параллельное функционирование, взаимодействие и распределение целей в сложной ИС.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 42-43; `Модели с иерархической структурой.docx`, с. 1-2.

27. Обобщенные модели. Концепции агрегатного и агентного моделирования и гибридного моделирования сложных систем

Обобщенной моделью сложной системы называют модель, в которой предмет моделирования представлен как многоуровневая структура взаимодействующих частей. Она строится до выбора конкретных уравнений или автоматов для отдельных отношений и задает общий каркас: уровни, агрегаты или агенты, интерфейсы, связи, локальные цели и параметры.

Агрегатное моделирование, или U-схемы, рассматривает систему как иерархию жестко связанных агрегатов, то есть блоков, устройств, узлов или других единиц. Локальные цели элементов подчинены общей цели функционирования системы, а динамика определяется глобальными правилами и законами взаимодействия.

Агентное моделирование, или A-схемы, рассматривает систему как децентрализованную. Ее общее поведение возникает из индивидуальной активности агентов. Связи между агентами не обязательно заданы заранее: они могут возникать ситуативно, когда одному агенту нужно взаимодействовать с другим, и исчезать после потери такой необходимости.

Гибридное моделирование используют, когда на разных уровнях одной системы нужны и U-схемы, и A-схемы. Например, инфраструктура может быть жестко агрегатной, а пользователи или программные компоненты - агентными. AnyLogic поддерживает такую смесь подходов, поэтому хорошо подходит для имитационного моделирования сложных ИС.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 43-45; `Тема_7_Имитационное_моделирование.pdf`, с. 10-12.

28. Понятие обобщенного моделирования сложных систем. Схемы обобщенного моделирования сложных информационных систем. Агрегатное моделирование структуры сложных систем (U-схемы)

Обобщенное моделирование сложной информационной системы - это построение многоуровневой формальной модели, где система декомпозируется на взаимодействующие объекты до уровня простых агрегатов. На верхних уровнях описывается структура и взаимодействие, а на нижних - параметры и отношения, которые можно интерпретировать D-, F-, P-, Q- или другими схемами.

В лекции для сложных систем названы три концептуальные схемы: агрегатное моделирование U-схемами, агентное моделирование A-схемами и гибридное моделирование. U-схемы нужны тогда, когда система рассматривается как иерархия жестко связанных агрегатов, работающих на общую цель.

Агрегатное моделирование структуры сложной системы означает, что элементы каждого уровня, кроме базового, инкапсулируют множество агрегатов нижнего уровня. Это могут быть устройства, блоки, узлы, подсистемы. На одном уровне агрегаты взаимодействуют обменом данными, а если агрегат оказывается сложным, он раскрывается на следующем уровне иерархии.

Смысл U-схемы - показать состав и связи информационных процессов как абстрактных преобразователей информации. После построения такой структуры внутреннее поведение простых агрегатов можно задавать конкретными математическими схемами, а затем проводить эксперименты с моделью всей системы.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 42-45.

Непрерывно-детерминированные модели (D-схемы)

29. Предназначение непрерывно-детерминированных моделей при разработке систем мониторинга и управления физическими объектами

Непрерывно-детерминированные модели применяют тогда, когда нужно описывать физический объект с параметрами, меняющимися во времени непрерывно и по закономерным правилам. Такие модели особенно важны для систем мониторинга и управления, потому что позволяют прогнозировать состояние объекта между измерениями и оценивать реакцию объекта на управляющие воздействия.

Если параметры объекта подчиняются известным физическим законам, то D-схема позволяет вычислять температуру, скорость, координату, давление, уровень и другие непрерывные величины во времени. На основе этого можно строить программные средства наблюдения, предсказания и выбора управляющего действия.

В материалах курса это прямо связывается с разработкой приложений реального времени для мониторинга состояния физического объекта. То есть модель нужна не только для объяснения процесса, но и как вычислительная основа системы, которая должна сопровождать реальный объект во времени.

Источники: `Тема_1_Непрерывно_детерминированные_модели_D_схемы.pdf`, с. 2-4; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 4-6.

30. Основные свойства D-схем. Вид моделей D-схем на примере моделирования процесса изменения температуры в плоской однородной пластине G. Нестационарные и стационарные решения

D-схемы описывают процессы, в которых параметры меняются непрерывно во времени, а случайный фактор в самой схеме отсутствует. Их основные свойства - непрерывное время, детерминированность, причинная зависимость текущего состояния от предыдущего состояния и входных воздействий, а также выражение отношений между параметрами через дифференциальные уравнения или их численные аналоги.

В лабораторной работе рассматривается тонкая квадратная теплопроводящая пластина G. Температура во внутренних точках пластины задается функцией $u(x, y, t)$, а изменение температуры описывается уравнением теплопроводности:

$$\frac{\partial u}{\partial t} - a\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = 0, \qquad (x, y) \in G.$$

В методичке для простоты берут $a = 1$. Начальное условие для внутренних точек имеет вид $u(x, y, 0) = 0$, а на границе задаются условия теплообмена или теплоизоляции в зависимости от варианта.

Нестационарное решение описывает переходный режим: распределение температуры еще меняется, то есть $\partial u/\partial t \neq 0$. Стационарное решение соответствует установившемуся режиму, когда температура в узлах сетки перестает меняться с заданной точностью. Для стационарного режима в непрерывной записи выполняется $\partial u/\partial t = 0$.

Источники: `Тема_1_Непрерывно_детерминированные_модели_D_схемы.pdf`, с. 3-12; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 4-6.

31. Аналитические и численные методы исследования моделей класса D-схем. Метод конечных разностей численного решения дифференциальных уравнений

Аналитические методы дают решение в виде явной функции или системы функций. Это возможно, когда уравнения, область и граничные условия достаточно просты. Преимущество аналитического решения в том, что оно показывает общую зависимость и не связано с конкретным шагом расчета.

Численные методы используют, когда точное решение получить трудно. Тогда непрерывная модель заменяется вычислительной схемой, которая дает приближенные значения параметров в узлах сетки и в отдельные моменты времени.

Метод конечных разностей заменяет производные конечными разностями. Например,

$$\frac{\partial u}{\partial t} \approx \frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t},$$

а вторую производную по координате можно записать как

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1,j}^{n} - 2u_{i,j}^{n} + u_{i-1,j}^{n}}{\Delta x^2}.$$

После такой замены дифференциальное уравнение превращается в рекуррентную расчетную схему. В задаче о пластине температура в каждом внутреннем узле на следующем временном слое вычисляется по температурам соседних узлов и по граничным условиям.

Источники: `Тема_1_Непрерывно_детерминированные_модели_D_схемы.pdf`, с. 11-22; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 4-6.

32. Особенности применения метода конечных разностей для исследования моделей класса D-схем. Проблема устойчивости решения разностных схем. Факторы, влияющие на устойчивость метода конечных разностей

При применении метода конечных разностей непрерывный объект заменяется дискретной сеткой по времени и, если нужно, по пространству. Поэтому результат всегда приближенный и зависит от выбора шагов дискретизации. Чем грубее сетка, тем быстрее расчет, но тем выше риск потерять точность и даже получить качественно неверное поведение модели.

Устойчивость разностной схемы означает, что малые погрешности исходных данных и промежуточных вычислений не приводят к неограниченному росту ошибки. Неустойчивая схема может давать численно "взрывающиеся" значения даже тогда, когда реальный процесс должен вести себя спокойно.

Для явной схемы теплопроводности типичное условие устойчивости связывает шаг времени с пространственными шагами:

$$a\Delta t\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2}\right) \le \frac{1}{2}.$$

При одинаковом шаге h по обеим координатам это дает правило $\Delta t \le h^2/(4a)$. Поэтому увеличение шага времени ускоряет расчет, но может нарушить устойчивость и репрезентативность результатов. В лабораторной работе этот риск проверяется экспериментально: требуется исследовать, как шаг дискретизации времени влияет на устойчивость вычисления матрицы температур.

Источники: `Тема_1_Непрерывно_детерминированные_модели_D_схемы.pdf`, с. 31-34; `ЛР №2 Моделирование непрерывно-детерминированных параметров физического объекта.pdf`, с. 4-6.

Дискретно-детерминированные модели (F-схемы)

33. Свойства F-схем. Характеристика автоматного отношения между параметрами простого агрегата

F-схемы используют для описания дискретно-детерминированного поведения, когда состояние системы меняется скачкообразно, в отдельные моменты или такты, а результат перехода однозначно определяется текущим состоянием и входом. Для таких схем характерны дискретность времени или дискретность смены состояний, конечное или счетное множество состояний и отсутствие случайности в функции переходов.

Автоматное отношение между параметрами простого агрегата означает, что новый набор значений параметров определяется правилом перехода из текущего состояния в следующее. То есть поведение задается не непрерывной формулой, а логикой "если текущее состояние таково и пришел такой вход, то агрегат переходит в такое-то состояние и формирует такой-то выход".

Такой способ описания особенно удобен для управляющей логики, протоколов, режимов работы устройств, обработчиков событий и других систем, где важны режимы, переключения и реакции на дискретные сигналы.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 2-4.

34. Конечные автоматы. Устойчивые состояния конечного автомата. Синхронные и асинхронные автоматы

Конечный автомат - это детерминированный автомат, у которого конечны множества входных знаков, выходных знаков и состояний. В лекции он задается как

$$F = \langle Z, X, Y, \varphi, \psi, z_0 \rangle,$$

где Z - конечное множество состояний, X - входной алфавит, Y - выходной алфавит, z_0 - начальное состояние, φ - функция переходов, ψ - функция выходов.

Устойчивым для входного знака x_i называется состояние z_k, в котором автомат при поступлении этого знака остается в том же состоянии: $\varphi(z_k, x_i) = z_k$, и формирует единственный соответствующий выход $\psi(z_k, x_i) = y_j$. Если для такого же состояния и входа выход неоднозначен, это считается ошибкой задания автомата.

Синхронный автомат за такт один раз считывает вход, формирует выход и переходит в новое состояние, а затем бездействует до следующего такта. Асинхронный автомат в пределах одного такта может многократно считывать один и тот же вход и менять состояние, пока не достигнет устойчивого состояния. Поэтому для асинхронного автомата принципиально важно, чтобы устойчивое состояние достигалось для каждого входного знака.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 5-6.

35. Автомат Мили. Автомат Мура

Автомат Мили - это конечный автомат, в котором выход зависит и от текущего состояния, и от текущего входа. Поэтому выходная реакция может измениться сразу при изменении входного сигнала, даже если состояние еще не сменилось.

Автомат Мура - это конечный автомат, в котором выход зависит только от текущего состояния. Из-за этого выход меняется только тогда, когда меняется само состояние автомата.

Формально различие выражается так:

- для автомата Мили: $y_k = \lambda(z_k, x_k)$;
- для автомата Мура: $y_k = \lambda(z_k)$.

Автомат Мили обычно экономичнее по числу состояний, а автомат Мура удобнее для анализа, потому что каждому состоянию соответствует определенный выход.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 6-7.

36. Способы задания конечных автоматов, особенности их применения

Конечные автоматы можно задавать несколькими равноправными способами: таблицей переходов и выходов, матрицей соединений, графом состояний, аналитически через функции переходов и выходов, а также алгоритмически или программным кодом.

Табличный способ удобен для строгого перечисления реакций автомата на все допустимые входы. Матрица соединений компактно показывает структуру переходов и удобна при формальном анализе. Граф состояний особенно нагляден и полезен для проектирования и обсуждения логики поведения. Аналитическая запись и программная реализация хороши тогда, когда автомат включается в вычислительную модель или программную систему.

Выбор способа зависит от цели. Для понимания логики чаще используют граф, для формальной спецификации - таблицу, для машинной обработки - матрицу или код.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 7-12.

37. Табличный способ задания конечного автомата. Таблица для задания автомата Мили

Табличный способ задания автомата Мили состоит в том, что для каждой пары "текущее состояние - вход" указывают сразу две вещи: следующее состояние и выходной символ. Обычно строки таблицы соответствуют текущим состояниям, а столбцы - входным сигналам.

Ячейка такой таблицы содержит запись вида

$$(z', y),$$

где z' - следующее состояние, а y - выход автомата при данном входе. Благодаря этому таблица одновременно задает функцию переходов δ и функцию выходов λ.

Особенность автомата Мили в том, что выход пишут именно в ячейке, потому что он зависит от входа. Это делает таблицу полной и удобной для пошагового моделирования поведения автомата.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 8-9.

38. Табличный способ задания конечных автоматов. Таблица для задания автомата Мура

Для автомата Мура таблица строится немного иначе, потому что выход зависит только от состояния. Поэтому обычно отдельно указывают выход, соответствующий каждому состоянию, а сама таблица переходов содержит только информацию о следующем состоянии для каждой пары "текущее состояние - вход".

То есть структура задания состоит из двух частей:

- таблица или столбец выходов $y = \lambda(z)$ для состояний;
- таблица переходов $z' = \delta(z, x)$.

Такое представление удобно тем, что видно устойчивое соответствие между состоянием и выходом. Если автомат вошел в состояние z_i, значит его выход уже известен и не зависит от того, какой вход вызвал этот переход.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 9.

39. Способ задания конечного автомата с помощью матрицы соединений. Матрица соединений автомата Мили

Матрица соединений описывает, какие переходы возможны между состояниями автомата и какими входами они вызываются. Для автомата Мили в матрице обычно отражают переходы из состояния в состояние, а также связывают их с соответствующими выходами, поскольку выход формируется на переходе.

Если строки и столбцы матрицы соответствуют состояниям, то элемент матрицы может содержать обозначение входа и выхода, при которых осуществляется переход из состояния строки в состояние столбца. То есть элемент матрицы имеет смысл дуги графа состояний.

Такое представление удобно для формального анализа связности автомата, достижимости состояний и структуры переходов, но по наглядности обычно уступает графу.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 9-11.

40. Способ задания конечного автомата с помощью матрицы соединений. Матрица соединений автомата Мура

Для автомата Мура матрица соединений прежде всего описывает переходы между состояниями по входам, а выходы указываются отдельно как свойства самих состояний. Поэтому элемент матрицы обычно отражает только условие перехода из одного состояния в другое.

Если автомат находится в состоянии z_i, его выход уже определен функцией $\lambda(z_i)$. Матрица соединений нужна для ответа на другой вопрос - в какое состояние попадет автомат при поступлении данного входа. Из-за такого разделения матрица автомата Мура обычно выглядит проще, чем для автомата Мили.

Основное достоинство этого способа - компактное представление структуры автомата при сравнительно большом числе состояний.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 10-11.

41. Способ задания конечных автоматов с помощью графа. Вид графа автомата Мили

Граф автомата Мили состоит из вершин и ориентированных дуг. Вершины соответствуют состояниям, а дуги - переходам между состояниями. Поскольку выход автомата Мили зависит от входа и формируется на переходе, на дуге обычно пишут пару вида "вход/выход".

Например, подпись на дуге x_j/y_k означает, что при входе x_j автомат переходит по данной дуге и формирует выход y_k. Именно поэтому граф автомата Мили наглядно показывает реакцию автомата как на уровне смены состояния, так и на уровне выдачи выхода.

Такой способ задания особенно удобен при анализе алгоритмов управления и последовательностей реакций на входные сигналы.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 11-12.

42. Способ задания конечных автоматов с помощью графа. Вид графа автомата Мура

Граф автомата Мура также состоит из вершин-состояний и ориентированных дуг-переходов, но выход связывается не с дугами, а с вершинами. Поэтому возле каждой вершины указывают состояние и соответствующий ему выход.

На дугах графа обычно подписывают только входные условия перехода. В результате граф автомата Мура читается так: автомат находится в состоянии, выдает привязанный к нему выход и при определенном входе переходит в новое состояние, вместе с которым изменяется и выход.

Такой граф удобен тем, что четко разделяет две сущности: режим работы автомата и условия переключения между режимами.

Источники: `Тема_2_Дискретно_детерминированные_модели_F_схемы.pdf`, с. 11-13.

Модель реагирующих систем

43. Моделирование простых реагирующих систем диаграммами состояний и переходов

Реагирующая система - это система, которая длительное время находится в некоторых состояниях и реагирует на события, сигналы, условия или истечение времени переходами в другие состояния. Простая реагирующая система обычно имеет ограниченное число режимов и сравнительно несложную логику переходов.

Диаграмма состояний и переходов задает такую систему наглядно: состояния показывают устойчивые режимы, а переходы - условия смены режима. В переходе можно указать событие, охранное условие, задержку по времени и действие, выполняемое при переходе. Пока переход не сработал, система остается в текущем состоянии.

В AnyLogic такой способ моделирования удобен для описания логики контроллеров, жизненного цикла объекта, режима работы устройства или поведения агента, у которого важны именно реакции на события, а не непрерывная динамика параметров.

Источники: `Тема_5_Реагирующие_системыR_схемы.pdf`, с. 2-4.

44. Моделирование сложных реагирующих систем. Иерархические, исторические и условные состояния в диаграммах состояний и переходов

В сложной реагирующей системе число состояний и переходов быстро растет, поэтому простая плоская диаграмма становится неудобной. Для борьбы со сложностью используют иерархию состояний, историю и условные разветвления.

Иерархическое состояние - это составное состояние, внутри которого есть вложенная диаграмма. Оно позволяет объединять группу близких режимов в один крупный блок. Историческое состояние запоминает, в каком вложенном подрежиме система находилась до выхода из составного состояния, и при возврате восстанавливает именно этот подрежим. Условные состояния и ветвления нужны тогда, когда дальнейший переход выбирается по логическому условию, а не по единственному заранее фиксированному пути.

Такие средства позволяют строить многоуровневые модели поведения без потери читаемости. По сути, это та же идея иерархической организации, что и при моделировании сложных систем в целом: сложное поведение раскладывается на уровни и локальные режимы.

Источники: `Тема_5_Реагирующие_системыR_схемы.pdf`, с. 5-12.

Дискретно-событийные стохастические модели (Р-схемы)

45. Свойства Р-схем. Математическая схема стохастического автомата (Р-автомата)

Р-схемы применяют для описания дискретных систем, у которых переходы между состояниями происходят по вероятностным правилам. В отличие от F-схем, здесь реакция системы при одинаковом состоянии и одинаковом входе может быть неоднозначной: одно и то же условие допускает несколько переходов с разными вероятностями.

Стохастический автомат, или Р-автомат, задается множествами входов, состояний и выходов, а также вероятностными законами переходов и, при необходимости, вероятностными законами формирования выходов. Поэтому в общей форме вместо обычной функции переходов используют распределение вероятностей по следующим состояниям.

Главное свойство Р-схемы состоит в том, что она описывает не одну фиксированную траекторию поведения, а вероятностное множество возможных траекторий. Анализ такой модели ведут через вероятности состояний, вероятности выходов и предельные характеристики процесса.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 2-6.

46. Вероятностный автомат Мили

Вероятностный автомат Мили - это Р-автомат, у которого для каждой пары входного знака и состояния (x_i, z_s) отдельно задаются распределение вероятностей переходов по состояниям и распределение вероятностей выходов. То есть имеются две таблицы: вероятности r_{is}^k перехода в состояние z_k и вероятности q_{is}^j появления выходного значения y_j.

Для каждой строки выполняются условия нормировки:

$$\sum_k r_{is}^k = 1, \qquad \sum_j q_{is}^j = 1.$$

Если дополнительно переход в новое состояние и выдача выходного значения независимы, то совместная вероятность пары (z_k, y_j) равна

$$b_{is}^{kj} = r_{is}^k q_{is}^j.$$

Именно такой Р-автомат в лекции называется стохастическим автоматом Мили. Если все вероятности в этих распределениях вырождаются в 0 и 1, он превращается в обычный детерминированный автомат Мили.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 7-8.

47. Вероятностный автомат Мура

Вероятностный автомат Мура - это Р-автомат, в котором выходное значение зависит только от состояния, достигнутого в текущем такте. Поэтому распределение переходов задается для пары (x_i, z_s), а распределение выходов - для состояния z_k.

Для состояний задаются вероятности q_k^j появления выходного значения y_j, причем

$$\sum_j q_k^j = 1.$$

Если событие перехода в состояние z_k и событие выдачи выхода y_j считаются независимыми, совместная вероятность записывается как

$$b_{is}^{kj} = r_{is}^k q_k^j.$$

Такой автомат удобен, когда режим системы сначала выбирается вероятностным переходом, а наблюдаемый выход определяется уже этим режимом. При вырождении вероятностей в значения 0 и 1 получается детерминированный автомат Мура.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 8-9.

48. Частично-детерминированные вероятностные автоматы. Y-детерминированный и Z-детерминированный P-автоматы

Частично-детерминированные P-автоматы занимают промежуточное место между полностью стохастическими и полностью детерминированными автоматами. В них одна часть поведения задается однозначно, а другая остается вероятностной.

Y-детерминированный P-автомат имеет детерминированный выход: для каждой пары (x_i, z_s) в таблице выходов только одна вероятность равна 1, остальные равны 0. При этом переход в следующее состояние может оставаться случайным.

Z-детерминированный P-автомат имеет детерминированный переход: для каждой пары (x_i, z_s) только одна вероятность перехода в состояние z_k равна 1. При этом выход может формироваться вероятностно.

Особый случай - Y_0-детерминированный автомат, у которого множество входных знаков состоит из одного символа. Тогда само значение входа несущественно, важен только факт такта срабатывания. Если вероятности переходов не зависят от предыстории попадания в текущее состояние, такой автомат становится марковским, а последовательность его состояний образует дискретную марковскую цепь.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 9-10.

49. Марковский случайный процесс. Y0-детерминированный P-автомат. Дискретная марковская цепь. Способы задания марковской цепи

Марковский случайный процесс обладает свойством отсутствия памяти: если известно текущее состояние, то распределение будущего состояния не зависит от того, как система пришла к этому состоянию. Прошлая история уже учтена в текущем состоянии.

Y_0-детерминированный P-автомат - это вероятностный автомат, в котором выход не играет роли. Остаются только состояния и вероятностные переходы между ними. Если переходы происходят по дискретным шагам времени, такая модель становится дискретной марковской цепью.

Марковская цепь задается множеством состояний Z, начальным распределением $\pi^{(0)}$ и матрицей переходных вероятностей

$$P = (p_{ij}), \qquad p_{ij} = P\{Z_{k+1} = j \mid Z_k = i\}, \qquad \sum_j p_{ij} = 1.$$

Распределение вероятностей после одного шага находится как $\pi^{(k+1)} = \pi^{(k)} P$. Задать цепь можно матрицей переходов, графом состояний с вероятностями на дугах или системой рекуррентных уравнений для вероятностей состояний.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 10-14.

50. Финальные вероятности. Нахождение финальных вероятностей для Y0-детерминированного P-автомата

Финальные вероятности - это предельные вероятности состояний, к которым стремится марковская цепь при большом числе шагов, если такой предел существует. Для Y_0-детерминированного P-автомата они показывают, какую долю времени в долгосрочном режиме система проводит в каждом состоянии.

Если $\pi = (\pi_1, ..., \pi_n)$ - вектор финальных вероятностей, то он удовлетворяет системе

$$\pi = \pi P, \qquad \sum_{i=1}^{n} \pi_i = 1, \qquad \pi_i \geq 0.$$

То есть финальное распределение не меняется после умножения на матрицу переходов. На практике из уравнений $\pi_j = \sum_i \pi_i p_{ij}$ вместе с нормировкой выбирают независимую систему линейных уравнений и решают ее.

В лекции подчеркивается: если пределы существуют и не зависят от начального состояния, они уже не являются функциями времени, а становятся константами. По смыслу финальная вероятность состояния - это средняя доля времени, которую автомат проводит в этом состоянии после длительного функционирования.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 15-17.

51. Разновидности марковских цепей. Поглощающая марковская цепь. Неприводимая марковская цепь

Марковские цепи различают по структуре графа переходов и по долгосрочному поведению. В лекционном материале для задач прототипирования программ особенно выделены поглощающие и неприводимые цепи.

Поглощающее состояние - это состояние i, для которого $p_{ii} = 1$. Попав в него, система уже не выходит. Поглощающая цепь содержит хотя бы одно такое состояние и хотя бы одно состояние, из которого поглощение достижимо. Такие цепи используют для моделирования завершения процесса, отказа, потери работоспособности, достижения цели.

Неприводимая цепь - это цепь, в которой любые два состояния сообщаются: из каждого состояния можно попасть в любое другое за конечное число шагов с положительной вероятностью. Для конечной неприводимой цепи все состояния принадлежат одному классу, поэтому долгосрочный анализ ведется для всей системы целиком, а не для отдельных изолированных частей.

Для неприводимой цепи основная характеристика - доля времени, или финальная вероятность, пребывания процесса в каждом состоянии. Анализ этих долей помогает находить слишком часто или слишком редко используемые модули и менять структуру программы.

Источники: `Тема_3_Дискретно_стохастические_модели_P_схемы.pdf`, с. 18-20.

Непрерывные событийно-стохастические модели (Q-схемы)

52. Понятие непрерывных событийно-стохастических систем. Свойства Q-схем

Непрерывная событийно-стохастическая система - это система, которая функционирует в непрерывном времени, но меняет состояние скачкообразно в случайные моменты наступления событий. Поэтому между событиями состояние может сохраняться, а сами переходы происходят мгновенно и вероятностно.

Q-схемы применяются прежде всего для процессов обслуживания. Их основные свойства таковы: непрерывное время, случайные моменты поступления и завершения событий, дискретное множество состояний системы, вероятностный характер переходов и возможность описания поведения через потоки событий и непрерывные марковские цепи.

В таких моделях естественно описываются очереди, каналы обслуживания, отказы, ожидание, захват и освобождение ресурсов. Именно поэтому Q-схемы особенно важны для анализа информационных систем как систем обслуживания.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 2-3.

53. Отношения между параметрами, задаваемые как процессы обслуживания. Основные понятия простейшего процесса обслуживания. Модели потоков событий

Если отношения между параметрами системы выражаются как прием, ожидание, обработка и выпуск заявок, то такую часть поведения удобно моделировать как процесс обслуживания. В этом случае параметры описывают поток заявок, очередь, состояние канала, длительность обслуживания, факт отказа, пропускную способность и другие характеристики.

В простейшем процессе обслуживания выделяют заявку, входной поток заявок, накопитель или очередь, канал обслуживания, выходной поток результатов и возможный отказ. Заявка либо немедленно обслуживается, либо ждет, либо получает отказ, если обслуживание невозможно.

Для задания процессов обслуживания нужно определить модели потоков событий - то есть каким образом появляются заявки и как распределены интервалы между событиями. От выбора модели потока зависит вся последующая вероятностная структура процесса.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 3-6, 10-11.

54. Модель абстрактного прибора обслуживания, характеристики эффективности его функционирования. Абсолютная и относительная пропускная способность прибора обслуживания

Абстрактный прибор обслуживания включает источник заявок, накопитель, один или несколько каналов обслуживания и механизм выдачи результата или отказа. Такой прибор описывает не конкретное устройство, а общую схему обработки заявок.

Эффективность его функционирования оценивают по времени ожидания, длине очереди, загрузке канала, вероятности отказа, числу обслуженных заявок и пропускной способности. Эти характеристики позволяют судить, насколько прибор справляется с потоком нагрузки.

Абсолютная пропускная способность - это среднее число заявок, реально обслуживаемых в единицу времени. Относительная пропускная способность - это доля заявок входного потока, которые получают обслуживание. Если обозначить интенсивность входного потока через λ, а относительную пропускную способность через q, то абсолютная пропускная способность равна $A = \lambda q$.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 3-9.

55. Моделирование потоков событий: регулярные потоки, потоки без последействий, потоки с ограниченным последействием, поток Пальма, стационарные и нестационарные потоки

Поток событий - это последовательность моментов времени, в которые наступают однотипные события: приходы заявок, завершения обслуживания, отказы, сбои, сигналы управления.

Регулярный поток имеет постоянные интервалы между событиями. Он детерминирован: если известен первый момент и период, известен весь поток.

Поток без последействия обладает свойством отсутствия памяти: вероятность событий в будущем не зависит от того, как события располагались в прошлом. Для стационарного ординарного потока без последействия получается простейший, или пуассоновский, поток.

Поток с ограниченным последействием допускает зависимость от предыстории, но эта зависимость учитывает только ограниченную часть прошлого или ограниченное число предыдущих событий.

Поток Пальма в лекции определяется как поток с ограниченным последействием, у которого интервалы между соседними событиями независимы и имеют одинаковый закон распределения. Простейший поток является частным случаем потока Пальма, когда интервалы имеют экспоненциальное распределение.

Стационарный поток имеет вероятностные характеристики, не зависящие от сдвига по оси времени. Нестационарный поток имеет меняющуюся интенсивность или другие характеристики, например дневные пики нагрузки в информационной системе.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 10-14.

56. Понятие ординарного потока событий. Определение интенсивности ординарного потока событий

Ординарным называют поток, в котором вероятность появления более одного события на очень малом интервале времени пренебрежимо мала по сравнению с вероятностью появления одного события. Проще говоря, события приходят поодиночке, а не пачками.

Интенсивность потока - это среднее число событий, приходящихся на единицу времени. Для стационарного потока интенсивность обозначают λ. Формально она может быть определена как предел отношения математического ожидания числа событий на малом интервале к длине этого интервала:

$$\lambda = \lim_{\Delta t \to 0} \frac{E[N(\Delta t)]}{\Delta t}.$$

Для ординарного стационарного потока вероятность одного события на малом интервале примерно равна $\lambda \Delta t$, а вероятность двух и более событий имеет более высокий порядок малости.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 14-15.

57. Понятие простейшего потока событий. Функция распределения вероятности интервалов времени между соседними событиями простейшего потока. Вероятностные характеристики простейшего потока

Простейший поток событий - это ординарный стационарный поток без последействия. Именно такой поток лежит в основе классических моделей массового обслуживания и эквивалентен пуассоновскому потоку событий.

Для простейшего потока интервалы времени между соседними событиями независимы и имеют экспоненциальное распределение с параметром λ:

$$F(t) = 1 - e^{-\lambda t}, \qquad t \geq 0.$$

Соответственно плотность равна $f(t) = \lambda e^{-\lambda t}$, математическое ожидание интервала - $1/\lambda$, дисперсия - $1/\lambda^2$. Число событий за интервал длины t имеет распределение Пуассона:

$$P\{N(t)=k\} = \frac{(\lambda t)^k}{k!} e^{-\lambda t}.$$

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 15-18.

58. Поток Эрланга k-го порядка, его вероятностные характеристики и свойства

Поток Эрланга k-го порядка получают из простейшего потока, если сохраняют каждое k-е событие, а остальные рассматривают как промежуточные фазы. Поэтому интервал между соседними событиями такого потока равен сумме k независимых экспоненциальных интервалов.

Если исходный простейший поток имеет интенсивность λ, то интервал T между событиями потока Эрланга имеет плотность

$$f_T(t) = \frac{\lambda^k t^{k-1} e^{-\lambda t}}{(k-1)!}, \qquad t \geq 0.$$

Его математическое ожидание и дисперсия равны

$$E[T] = \frac{k}{\lambda}, \qquad D[T] = \frac{k}{\lambda^2}.$$

Интенсивность получившегося потока равна λ/k, потому что события сохраняются реже. Иногда поток Эрланга задают не через исходную интенсивность, а через желаемую интенсивность ν самого потока; тогда в плотности используют параметр $k\nu$, чтобы $E[T] = 1/\nu$.

При $k = 1$ поток Эрланга совпадает с простейшим потоком. При росте k относительный разброс интервалов уменьшается, и поток становится более регулярным.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 18-21.

59. Моделирование процесса обслуживания непрерывной марковской цепью. Вероятности состояний непрерывной марковской цепи. Уравнения Колмогорова. Финальные вероятности состояний

Процесс обслуживания можно описывать непрерывной марковской цепью, если состояние системы меняется в непрерывном времени, а будущее зависит только от текущего состояния. В СМО состояние часто задают числом заявок в системе, числом занятых каналов или парой "очередь - обслуживающие каналы".

Для непрерывной марковской цепи задают интенсивности переходов q_{ij} из состояния i в состояние j. Вероятности состояний $p_i(t)$ удовлетворяют уравнениям Колмогорова. В общем виде:

$$\frac{dp_i(t)}{dt} = \sum_{j \neq i} p_j(t) q_{ji} - p_i(t) \sum_{j \neq i} q_{ij}.$$

Первая сумма - приток вероятности в состояние i, второй член - отток вероятности из него.

Финальные, или стационарные, вероятности p_i находят из условий $dp_i(t)/dt = 0$ и нормировки $\sum_i p_i = 1$. Полученные вероятности позволяют вычислить вероятность отказа, среднюю длину очереди, загрузку каналов, пропускную способность и другие характеристики процесса обслуживания.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 21-27.

60. Моделирование и характеристики одноканального процесса обслуживания с отказами и простейшими потоками поступления заявок и обслуживания

Одноканальная СМО с отказами имеет один канал обслуживания и не имеет очереди. Если канал свободен, пришедшая заявка начинает обслуживаться. Если канал занят, заявка получает отказ и сразу покидает систему. Такая модель обозначается $M/M/1/1$.

Состояния системы: 0 - канал свободен, 1 - канал занят. Пусть λ - интенсивность поступления заявок, μ - интенсивность обслуживания, $\rho = \lambda/\mu$ - приведенная нагрузка. Стационарные вероятности равны

$$p_0 = \frac{1}{1+\rho}, \qquad p_1 = \frac{\rho}{1+\rho}.$$

Для простейшего входного потока вероятность отказа равна вероятности застать канал занятым:

$$P_{\text{отк}} = p_1 = \frac{\rho}{1+\rho}.$$

Относительная пропускная способность

$$q = 1 - P_{\text{отк}} = p_0 = \frac{1}{1+\rho},$$

а абсолютная пропускная способность

$$A = \lambda q = \frac{\lambda}{1+\rho} = \mu p_1.$$

При росте нагрузки вероятность отказа увеличивается, а абсолютная пропускная способность приближается к предельной способности канала.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 27-29.

61. Моделирование и характеристики многоканальной СМО с отказами и простейшими потоками поступления заявок и обслуживания. Формулы Эрланга

Многоканальная СМО с отказами имеет n параллельных каналов и не имеет очереди. Если при поступлении заявки хотя бы один канал свободен, заявка принимается. Если заняты все n каналов, заявка теряется. Модель обозначается $M/M/n/n$.

Состояние системы задают числом занятых каналов $k = 0, 1, ..., n$. При приведенной нагрузке $\rho = \lambda/\mu$ стационарные вероятности имеют вид

$$p_k = \frac{\rho^k/k!}{\sum_{i=0}^{n} \rho^i/i!}, \qquad k = 0, 1, ..., n.$$

Вероятность отказа равна вероятности состояния n, то есть задается формулой Эрланга B:

$$B(n, \rho) = p_n = \frac{\rho^n/n!}{\sum_{i=0}^{n} \rho^i/i!}.$$

Относительная пропускная способность $q = 1 - B(n, \rho)$, абсолютная $A = \lambda q$, среднее число занятых каналов равно A/μ. Формулы Эрланга показывают, как добавление каналов снижает потери заявок при той же нагрузке.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 29-32.

62. Моделирование и характеристики одноканальной СМО с ожиданием и простейшими потоками поступления заявок и обслуживания

Одноканальная СМО с ожиданием имеет один канал и очередь неограниченной или достаточно большой емкости. Если канал занят, заявка не теряется, а становится в очередь. Классическая модель - $M/M/1$.

Для существования стационарного режима нужно условие

$$\rho = \frac{\lambda}{\mu} < 1.$$

Если $\lambda \geq \mu$, канал в среднем не успевает обслуживать поток, и очередь растет без устойчивого предела.

В стационарном режиме вероятность того, что в системе находится n заявок, равна

$$p_n = (1-\rho)\rho^n, \qquad n = 0, 1, 2, ...$$

Среднее число заявок в системе

$$L = \frac{\rho}{1-\rho},$$

средняя длина очереди

$$L_q = \frac{\rho^2}{1-\rho}.$$

По формулам Литтла среднее время пребывания заявки в системе $W = L/\lambda = 1/(\mu - \lambda)$, а среднее время ожидания в очереди $W_q = L_q/\lambda = \rho/(\mu - \lambda)$. При $\rho \to 1$ очередь и задержки резко возрастают.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 33-35.

63. Понятие замкнутых СМО. Пример моделирования замкнутой СМО с n простейшими входными потоками заявок и одним каналом обслуживания

Замкнутая СМО отличается от открытой тем, что общее число источников заявок конечно. Заявки не приходят из бесконечной внешней среды: после обслуживания источник через некоторое время снова может породить заявку. Поэтому интенсивность поступления зависит от того, сколько источников сейчас свободно.

Пример - система с n источниками и одним каналом обслуживания. Если в системе уже находится k заявок, то свободны $n-k$ источников. При простейших потоках от каждого свободного источника суммарная интенсивность рождения новой заявки равна

$$\lambda_k = (n-k)\lambda, \qquad k = 0, 1, ..., n-1.$$

Интенсивность обслуживания при $k > 0$ равна μ. Состояние удобно задавать числом заявок k в системе. Для стационарных вероятностей такой цепи рождения-гибели можно записать

$$p_k = p_0 \frac{n!}{(n-k)!} \left(\frac{\lambda}{\mu}\right)^k, \qquad k = 0, 1, ..., n,$$

где p_0 определяется из нормировки $\sum_{k=0}^{n} p_k = 1$.

Замкнутость ограничивает максимальную длину очереди числом источников и делает поток поступления зависящим от состояния системы.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 36-38.

64. Обобщенные СМО. Особенности их моделирования

Обобщенными называют СМО, в которых ослабляют классические предположения простых моделей. В таких системах могут быть неоднородные потоки заявок, несколько классов приоритетов, произвольные распределения времени обслуживания, сети из взаимосвязанных узлов, ограниченные буферы, маршрутизация, возвраты заявок, расписания ресурсов и другие усложнения.

Особенность их моделирования в том, что аналитические формулы либо сильно усложняются, либо вообще отсутствуют. Поэтому приходится использовать имитационное моделирование, где структура системы собирается из блоков обслуживания, очередей, ресурсов, маршрутизаторов и сборщиков статистики.

Именно в обобщенных СМО проявляется практическая ценность AnyLogic: модель можно строить по структуре реального процесса и затем статистически оценивать ее характеристики без требования получить замкнутое аналитическое решение.

Источники: `Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf`, с. 38-41; `ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf`, с. 8-18, 29-46; `Блок Wait.pdf`, с. 1-6; `Блоки TimeMeasure_selectOutput.pdf`, с. 1-3.

Теоретические основы анализа стохастических моделей

65. Предельные теоремы теории вероятностей

Предельные теоремы описывают, что происходит со случайными величинами, частотами и средними при большом числе наблюдений. Для статистического и имитационного моделирования это основа: один прогон случайной модели может быть нетипичным, а серия прогонов дает устойчивые оценки.

Ключевая идея закона больших чисел: выборочное среднее независимых одинаково распределенных величин при росте числа наблюдений приближается к математическому ожиданию. В вероятностной форме:

$$P\left\{\left|\frac{1}{n}\sum_{i=1}^{n} X_i - E[X]\right| > \varepsilon\right\} \to 0.$$

Теорема Бернулли применяет эту идею к частоте события: при большом числе независимых испытаний относительная частота события приближается к его вероятности. Центральная предельная теорема уточняет форму случайных отклонений: нормированная сумма большого числа независимых слагаемых часто имеет распределение, близкое к нормальному.

В моделировании эти теоремы объясняют, почему можно оценивать вероятность отказа, среднее время обслуживания, среднюю длину очереди и доверительные интервалы по накопленной статистике экспериментов.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 3-6.

66. Теоретические основы метода статистического моделирования. Предельные теоремы Бернулли, Чебышева. Центральная предельная теорема

Метод статистического моделирования основан на многократном воспроизведении случайного процесса и последующей статистической обработке наблюдений. В имитационной модели это означает: многократно сгенерировать значения экзогенных случайных параметров, преобразовать их моделью в эндогенные характеристики и по накопленной выборке оценить вероятности, средние значения и другие показатели.

Теорема Бернулли обосновывает статистическую оценку вероятности события. Если проводится N независимых испытаний, а событие A имеет вероятность $p(A)$, то его относительная частота m_N сходится по вероятности к $p(A)$:

$$\forall \varepsilon > 0: \quad \lim_{N\to\infty} P\{|m_N - p(A)| < \varepsilon\} = 1.$$

Теорема Чебышева обосновывает оценку математического ожидания. Если $x_1, x_2, \ldots$ - попарно независимые одинаково распределенные случайные величины с математическим ожиданием a и конечной дисперсией, то среднее арифметическое стабилизируется около a:

$$\forall \varepsilon > 0: \quad \lim_{N\to\infty} P\left\{\left|\frac{1}{N}\sum_{i=1}^{N} x_i - a\right| < \varepsilon\right\} = 1.$$

Центральная предельная теорема объясняет, почему суммы и средние большого числа независимых одинаково распределенных величин часто приближаются к нормальному закону. В лекции для среднего $X_N = \frac{1}{N}\sum x_i$ указано: при больших N оно приближается к нормальному распределению с математическим ожиданием a и дисперсией σ^2/N.

Практический вывод такой: чем больше прогонов стохастической модели, тем устойчивее оценки, но рост точности требует резко большего числа вычислений. Поэтому статистическое моделирование всегда связано с компромиссом между точностью и временем эксперимента.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 4-6.

67. Моделирование непрерывных случайных величин, равномерно распределенных на отрезке [0,1]. Понятие квазинепрерывной случайной величины

Равномерная случайная величина U на $[0,1]$ - базовый источник случайности в статистическом моделировании. Если умеем получать такие значения, из них можно строить дискретные события, экспоненциальные интервалы, нормальные величины и другие распределения.

Для идеальной непрерывной величины плотность равна единице на $[0,1]$ и нулю вне этого отрезка, поэтому

$$P\{a \le U \le b\} = b - a, \qquad 0 \le a \le b \le 1.$$

Ее основные характеристики:

$$M[U] = \frac{1}{2}, \qquad D[U] = \frac{1}{12}.$$

Компьютерный генератор на самом деле не выдает бесконечно много значений. В лекции непрерывная величина заменяется n-разрядным дискретным аналогом, который принимает 2^n равномерно расположенных значений

$$x_k = \frac{k}{2^n - 1}, \qquad k = 0, 1, \ldots, 2^n - 1.$$

Такую величину называют квазинепрерывной. Ее математическое ожидание совпадает с $1/2$, а дисперсия при росте n стремится к $1/12$, поэтому при большой разрядности она практически мало отличается от непрерывной равномерной величины.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 10-12.

68. Способы получения и виды генераторов случайных чисел. Физические методы моделирования равномерно распределенных случайных чисел

Случайные числа получают физическими, табличными и алгоритмическими методами. Базовый генератор в курсе рассматривается как источник равномерно распределенных значений на $[0,1]$, потому что из него затем строят генераторы для других законов распределения.

Физический генератор опирается на реальный случайный процесс. В лекции разобран пример с подбрасыванием монеты: орел кодируется единицей, решка - нулем. Полученная последовательность двоичных цифр затем используется для построения числа на отрезке $[0,1]$.

Первый способ - последовательное деление отрезка пополам. Каждая очередная двоичная цифра выбирает левую или правую половину текущего интервала; после нескольких испытаний получается узкий интервал, любое число из которого можно принять как сгенерированное значение с заданной точностью.

Второй способ - собрать двоичные цифры в группы, например в триады, перевести их в восьмеричные цифры и записать дробь вида $0.xxx_8$, а затем при необходимости перевести ее в десятичную систему. В лекции также указан аппаратный генератор случайного шума электрического напряжения, после обработки которого получают случайную последовательность нулей и единиц.

Плюс физического метода - он использует действительно случайный источник. Минусы - техническая сложность, ограниченная скорость и слабая воспроизводимость: один и тот же эксперимент трудно повторить с той же последовательностью случайных чисел.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 13-15.

69. Виды генераторов случайных чисел. Табличные методы моделирования равномерно распределенных случайных чисел

Табличный метод использует заранее подготовленные таблицы случайных цифр или чисел. При моделировании из таблицы последовательно берут группы цифр фиксированной длины и переводят их в значения на интервале $[0,1)$.

Например, четырехзначное число от 0000 до 9999 можно разделить на 10000 и получить значение $U \in [0,1)$. Если таблица качественная, такие значения приближенно ведут себя как равномерные.

Достоинство табличного метода - независимость от программного алгоритма и простота для учебных расчетов. Недостатки - конечный объем таблицы, неудобство при больших сериях экспериментов, риск повторов и сложность автоматизации. Поэтому в современных вычислительных моделях табличный метод почти вытеснен алгоритмическими генераторами.

Таблицы полезны как исторический и учебный способ: они показывают, что моделирование случайности сводится к получению равномерной базовой последовательности, из которой затем строятся другие распределения.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 15-16.

70. Алгоритмические методы моделирования равномерно распределенных случайных чисел. Линейный конгруэнтный метод

Алгоритмический генератор строит псевдослучайную последовательность по детерминированной рекуррентной формуле

$$r_{i+1} = f(r_i).$$

Она выглядит случайной, но при известном начальном значении полностью воспроизводима. В лекции подчеркивается, что такие последовательности имеют период: рано или поздно возникает цикл повторения.

Простой исторический пример - метод середины квадрата Неймана: начальное число возводят в квадрат, из середины дробной части берут очередные разряды и получают следующее число. Метод прост, но качество сильно зависит от стартового числа.

Более устойчивый классический пример - линейный конгруэнтный метод:

$$X_{n+1} = (aX_n + c) \bmod m.$$

Здесь m - модуль, a - множитель, c - приращение, X_0 - начальное значение. В обозначениях лекции та же идея записана как $r_{i+1} = \mathrm{mod}(kr_i + b, M)$. Равномерное число на $[0,1)$ получают нормировкой результата по модулю.

Качество генератора зависит от периода и статистических свойств последовательности. Модуль выбирают достаточно большим, часто связанным со степенью двойки или с большим простым числом, а параметры рекуррентного преобразования подбирают так, чтобы период был длинным и не возникали заметные корреляции.

Плюсы метода - простота, скорость и воспроизводимость. Минусы - детерминированность и плохое качество при неудачном выборе параметров. Поэтому в практических системах используют тщательно проверенные генераторы, а не произвольные значения a, c и m.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 16-20.

71. Моделирование случайного числа, равномерно распределенного в заданном интервале

Если U равномерно распределена на $[0,1]$, то равномерную величину X на интервале $[a,b]$ получают линейным преобразованием:

$$X = a + (b-a)U.$$

Это преобразование растягивает единичный отрезок до нужного диапазона и сохраняет равномерность. Для любого подинтервала $[c,d] \subseteq [a,b]$ вероятность попадания равна отношению длин:

$$P\{c \le X \le d\} = \frac{d-c}{b-a}.$$

Если генератор выдает значения на полуинтервале $[0, 1)$, то результат будет лежать на $[a, b)$. В имитационных моделях это обычно несущественно, потому что вероятность ровно граничного значения у непрерывной величины равна нулю.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 21-22.

72. Моделирование полной группы несовместных событий

Пусть события $A_1, A_2, \dots, A_n$ несовместны и образуют полную группу, а их вероятности равны $p_1, p_2, \dots, p_n$, причем

$$\sum_{i=1}^{n} p_i = 1.$$

Для моделирования генерируют $U \sim U[0, 1]$ и разбивают отрезок $[0, 1]$ на интервалы длиной p_i. Если U попало в интервал события A_i, считают, что произошло именно A_i.

На практике используют накопленные вероятности:

$$S_i = p_1 + p_2 + \dots + p_i.$$

Событие A_i выбирается, если $S_{i-1} < U \leq S_i$ при $S_0 = 0$. Этот же прием используется для дискретных распределений, вероятностной маршрутизации и выбора выхода в блоках, где дальнейшее направление задается вероятностью.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 20-21.

73. Моделирование непрерывной случайной величины с заданным законом распределения. Метод обратной функции

Метод обратной функции строит случайную величину с заданной функцией распределения $F(x)$ из равномерной величины $U \sim U[0, 1]$. Если F строго возрастает и обратима, берут

$$X = F^{-1}(U).$$

Тогда X имеет функцию распределения F. Если F не строго обратима, используют обобщенную обратную функцию:

$$F^{-1}(u) = \inf\{x : F(x) \geq u\}.$$

Пример - экспоненциальное распределение с параметром λ. Его функция распределения $F(x) = 1 - e^{-\lambda x}$, поэтому

$$X = -\frac{1}{\lambda} \ln(1 - U).$$

Так как $1 - U$ также равномерна на $[0, 1]$, часто пишут эквивалентно $X = -\ln U/\lambda$. Метод хорош, когда обратная функция известна или легко вычисляется.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 22-23.

74. Моделирование гауссовской случайной величины

В лекции гауссовская случайная величина моделируется через центральную предельную теорему. Идея такая: сумма большого числа независимых одинаково распределенных случайных величин имеет распределение, близкое к нормальному.

Пусть ξ_i - независимые равномерные величины на $[0, 1]$. Для них

$$M[\xi_i] = \frac{1}{2}, \qquad D[\xi_i] = \frac{1}{12}.$$

Если сложить n таких величин и нормировать сумму, получим приближение к стандартному нормальному распределению:

$$\eta = \sqrt{\frac{12}{n}} \sum_{i=1}^{n} \left(\xi_i - \frac{1}{2}\right).$$

В практическом варианте часто берут $n = 12$. Тогда формула упрощается:

$$\eta = \sum_{i=1}^{12} \xi_i - 6.$$

Такая величина имеет примерно нулевое математическое ожидание и единичную дисперсию, то есть приближает $N(0,1)$. Чтобы получить нормальную величину с математическим ожиданием a и среднеквадратическим отклонением σ, используют преобразование $X = a + \sigma\eta$.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 24.

75. Моделирование непрерывной случайной величины с заданным законом распределения. Метод ступенчатой аппроксимации

Метод ступенчатой аппроксимации применяют для моделирования ограниченной непрерывной случайной величины с заданной плотностью $f_\xi(x)$. Сначала область определения плотности разбивают на n непересекающихся интервалов одинаковой ширины Δ.

На каждом интервале плотность заменяют прямоугольной ступенью высоты h_i. В лекции высота выбирается так, чтобы ступень находилась под графиком плотности и касалась его. После этого непрерывную величину приближенно заменяют дискретной величиной, которая принимает значения $x_1, x_2, ..., x_n$.

Вероятность выбрать значение из i-го интервала пропорциональна площади соответствующего прямоугольника:

$$P(x_i) = \frac{\Delta h_i}{\sum_{k=1}^{n} \Delta h_k} = \frac{h_i}{\sum_{k=1}^{n} h_k}.$$

Далее выполняют нормировку, чтобы сумма вероятностей была равна единице, и моделируют уже дискретную случайную величину: отрезок $[0,1]$ разбивают на интервалы длиной $P(x_i)$, генерируют равномерное число и по попаданию в интервал выбирают соответствующее x_i.

Точность метода зависит от числа интервалов: чем их больше, тем ближе ступенчатая аппроксимация к исходной плотности. Недостатки тоже важны: метод применим прежде всего к ограниченным областям и заменяет все значения интервала одним представительным значением, поэтому часть непрерывности теряется.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 24-26.

76. Моделирование непрерывной случайной величины с заданным законом распределения. Метод усечения

В задачах моделирования под методом усечения часто понимают метод отбора, или отбраковки: из простой области генерируют кандидатов и отсекают те, которые не попали под график нужной плотности. Метод полезен, когда плотность $f(x)$ известна, но обратную функцию распределения найти трудно.

Для ограниченного интервала $[a, b]$ выбирают число M, такое что $f(x) \le M$ на всем интервале. Затем генерируют две независимые равномерные величины: кандидат

$$X^* = a + (b - a)U_1$$

и высоту $Y = MU_2$. Если $Y \le f(X^*)$, кандидат принимают как значение X; если $Y > f(X^*)$, его отбрасывают и повторяют попытку. Геометрически это означает равномерное бросание точек в прямоугольник и усечение области над графиком плотности.

В более общем варианте используют огибающую плотность $g(x)$ и константу c, для которых $f(x) \le cg(x)$. Кандидат X^* генерируют по g, а принимают с вероятностью

$$\frac{f(X^*)}{cg(X^*)}.$$

Метод точен при правильной огибающей, но может быть медленным, если принимается малая доля кандидатов. Если в конкретном контексте под усечением понимают буквальное отбрасывание хвостов распределения, то сначала ограничивают область значений, перенормируют плотность на оставшемся интервале и затем моделируют уже усеченный закон.

Источники: `Тема_6_Моделирование_и_анализ_стохастических_параметров.pdf`, с. 26-28.

Имитационное моделирование систем

77. Общая характеристика метода имитационного моделирования, его отличие от построения математических моделей

Имитационное моделирование - это метод исследования систем путем построения исполняемой модели, которая объектно имитирует структуру и реальную работу предмета моделирования во времени. Такая модель не обязана сводиться к одной компактной формуле. Она может включать алгоритмы, блоки обслуживания, очереди, ресурсы, события, агентов, сбор статистики и визуализацию.

Главное отличие от построения математической модели состоит в форме результата и способе анализа. Математическая модель обычно выражается системой уравнений, функций или строгих отношений между параметрами, а затем исследуется аналитически или численно. Имитационная модель ориентирована на воспроизведение поведения системы в ходе прогона и на получение характеристик через компьютерный эксперимент.

На практике эти подходы не противостоят друг другу. Имитационная модель часто строится на основе математических зависимостей, но дополняет их структурой процессов, случайностью, логикой маршрутизации и возможностью прямого экспериментирования.

Источники: `Тема_7_Имитационное_моделирование.pdf`, с. 2-8, 12-14.

78. Этапы имитационного моделирования: концептуальное моделирование, реализация модели, калибровка и идентификация модели, компьютерный эксперимент

Построение имитационной модели в лекции делится на несколько этапов.

Концептуальное моделирование - это структуризация модели: выделение уровней иерархии, подсистем, элементарных компонентов и связей между ними. На этом этапе формулируют цель моделирования, решают, какие процессы нужно включить в модель, какие характеристики учитывать, а от каких деталей можно абстрагироваться.

Реализация модели средствами моделирования - это перевод концептуальной схемы в компьютерную модель: элементы системы, связи, параметры, переменные, события, диаграммы, блоки, статистика и визуализация получают конкретное представление в выбранной среде.

Калибровка и идентификация модели связаны со сбором данных и подбором значений параметров, которые должны быть введены в модель. Затем проводится валидация: выходные характеристики проверяют на режимах, где реальное или ожидаемое поведение системы заранее известно.

Компьютерный эксперимент - это выполнение модели при различных факторах и регистрация характеристик поведения. Более сложные эксперименты включают анализ чувствительности, сравнение управляющих воздействий и поиск оптимальных параметров.

Источники: `Тема_7_Имитационное_моделирование.pdf`, с. 9-10.

79. Направления и инструментальные средства имитационного моделирования. Общая характеристика системы имитационного моделирования AnyLogic

В лекции выделены четыре направления имитационного моделирования: моделирование динамических систем, дискретно-событийное моделирование, системная динамика и агентное моделирование. Выбор направления зависит от того, что является главным в системе: уравнения непрерывной динамики, потоки заявок и ресурсы, потоковые диаграммы накопителей и потоков или автономное поведение агентов.

Инструментальные средства имитационного моделирования должны обеспечивать построение модели, управление временем, запуск экспериментов, сбор статистики и визуализацию результатов. AnyLogic как раз объединяет эти возможности в одной среде.

Общая характеристика AnyLogic такова: это многоподходная система моделирования, поддерживающая агентные диаграммы, процессные блоки, элементы системной динамики, события, статистические объекты, различные типы экспериментов и средства презентации выполнения модели. Поэтому она удобна не только для расчета, но и для наглядного исследования поведения системы.

Источники: Тема_7_Имитационное_моделирование.pdf, с. 10-12; Практические основы компьютерного моделирования систем в AnyLogic.pdf, с. 14-20.

80. Модели и управление временем в системе моделирования AnyLogic

В AnyLogic время является одной из базовых характеристик модели. В зависимости от типа модели и выбранного режима оно может течь непрерывно, дискретно по тикам или продвигаться скачками от события к событию. Это позволяет поддерживать разные формализмы в одной среде.

В материалах курса особенно выделяются режимы реального и виртуального времени. Реальное время полезно для наглядной анимации и наблюдения за процессом, а виртуальное - для быстрого выполнения модели без привязки к физическому течению времени. Для непрерывных моделей также важен шаг дискретизации, влияющий на точность вычислений.

Управление временем в AnyLogic включает выбор единицы модельного времени, настройку скорости выполнения, длительности эксперимента, условий автоматического завершения и, при необходимости, переключение между наглядностью и скоростью расчета.

Источники: Тема_0_Введение_Агрегатное_моделирование_ИС.pdf, с. 23-27; Практические основы компьютерного моделирования систем в AnyLogic.pdf, с. 14-20; ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf, с. 22-24, 40.

81. Назначение статических и динамических параметров при моделировании агентов

При моделировании агентов статические параметры нужны для задания устойчивых свойств конкретного агента или условий эксперимента. К ним относятся идентификационные признаки, постоянные коэффициенты, настройки, емкости, фиксированные времена, характеристики среды и другие значения, которые не должны самопроизвольно меняться в ходе прогона.

Динамические параметры описывают текущее состояние агента и меняются по мере его функционирования. Это могут быть координаты, текущий режим, накопленные значения, время прихода, длина очереди, вероятность отказа, занятость ресурса и так далее.

Иначе говоря, статические параметры отвечают на вопрос, каким агент создан и в каких условиях он работает, а динамические - в каком состоянии он находится сейчас и как это состояние развивается.

Источники: Тема_0_Введение_Агрегатное_моделирование_ИС.pdf, с. 28-29; ЛР №1 Моделирование динамических систем в AnyLogic.pdf, с. 5, 13; ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf, с. 33-38.

82. Процессно-ориентированное моделирование процессов обслуживания в AnyLogic

Процессно-ориентированное моделирование в AnyLogic строится вокруг процессной диаграммы, по которой движутся агенты-заявки. Каждый блок диаграммы соответствует некоторой операции: создание заявки, ожидание, обслуживание, выбор маршрута, уничтожение, захват или освобождение ресурса, измерение времени пребывания и так далее.

Такой подход особенно удобен для процессов обслуживания, потому что позволяет прямо отразить логику прохождения клиента или заявки через систему. Структура очередей, емкости, распределения времени обслуживания, количество ресурсов и вероятностная маршрутизация задаются через свойства блоков.

В результате модель получается близкой к реальному процессу и легко поддается статистическому анализу: можно измерять длину очередей, время пребывания, загрузку ресурсов, вероятность отказа, абсолютную и относительную пропускную способность.

Источники: Тема_4_Непрерывные_событийно_стохастические_модели_Q_схемы.pdf, с. 3-9; ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf, с. 8-18, 19-46; Блок Wait.pdf, с. 1-6; Блоки TimeMeasure_selectOutput.pdf, с. 1-3.

83. Понятие эксперимента в AnyLogic. Параметры эксперимента. Типы экспериментов

Эксперимент в AnyLogic - это специальный объект проекта, который определяет, как именно будет выполняться модель. Сам по себе агент описывает структуру и поведение системы, а эксперимент задает условия прогона, параметры запуска, длительность, скорость времени, способ визуализации и сбор результатов.

Параметры эксперимента обычно включают время начала и окончания, единицу и режим модельного времени, наследуемые параметры модели, настройки презентации и правила завершения прогона. Важно, что эксперимент может наследовать параметры модели, позволяя изменять их между прогонами без перестройки самой диаграммы.

В курсах по AnyLogic прямо упоминаются по меньшей мере такие типы экспериментов: простой эксперимент, оптимизация и варьирование параметров. Каждый тип решает свою задачу: обычный прогон, поиск лучших значений и исследование влияния параметров.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 38-41; `Тема_7_Имитационное_моделирование.pdf`, с. 10; `Практические основы компьютерного моделирования систем в AnyLogic.pdf`, с. 15-20; `ЛР №1 Моделирование динамических систем в AnyLogic.pdf`, с. 13.

84. Категории параметров, используемых для спецификации свойств предмета моделирования в AnyLogic: параметры, переменные, динамические переменные, накопители

Параметры в AnyLogic используют для задания конфигурационных и обычно устанавливаемых значений модели. Они удобны для тех величин, которые нужно менять между прогонами или передавать в агент при его создании.

Переменные служат для хранения текущих вычисляемых значений, которые могут изменяться в ходе работы модели программным кодом или логикой блоков. Это общая категория изменяемых атрибутов.

Динамические переменные применяются в непрерывных моделях, когда значение должно изменяться во времени по заданному выражению или по связи с другими величинами. Они ближе к функциональным зависимостям и непрерывной динамике.

Накопители, или stocks, используют в системной динамике для представления величин, которые изменяются за счет потоков или по дифференциальным уравнениям. Именно накопитель хранит текущее значение состояния, а его изменение вычисляется по производной или суммарному потоку.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 20-29; `Практические основы компьютерного моделирования систем в AnyLogic.pdf`, с. 11-14, 18-20; `ЛР №1 Моделирование динамических систем в AnyLogic.pdf`, с. 13.

85. Оптимизационные эксперименты с моделями в AnyLogic

Оптимизационный эксперимент в AnyLogic предназначен для автоматического поиска наилучших значений параметров модели по выбранному критерию. Пользователь задает параметры, которые разрешено менять, ограничения на них и целевую функцию, например минимум времени, максимум пропускной способности или минимум вероятности отказа.

После этого AnyLogic выполняет серию прогонов модели, сравнивает результаты и направляет поиск в сторону улучшения целевого показателя. Таким образом модель начинает использоваться не просто для наблюдения, а как инструмент вычислительного выбора решений.

В учебной логике курса оптимизационный эксперимент особенно важен там, где нужно подобрать управляемые параметры объекта так, чтобы добиться требуемого поведения системы. Это относится и к непрерывным физическим моделям, и к моделям обслуживания.

Источники: `Тема_0_Введение_Агрегатное_моделирование_ИС.pdf`, с. 40; `Тема_7_Имитационное_моделирование.pdf`, с. 10; `Практические основы компьютерного моделирования` систем в AnyLogic.pdf, с. 15; ЛР №1 Моделирование динамических систем в AnyLogic.pdf, с. 10, 13; ЛР №3 Моделирование непрерывных событийно-стохастических систем.pdf, с. 47-48.